Notice of Allowability	Application No.	Applicant(s)	
	10/091,415	NEDBAL ET AL.	
	Examiner	Art Unit	
	Insun Kang	2193	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☒ This communication is responsive to *1/16/2007 and 1/22/2007*.

2. ☒ The allowed claim(s) is/are *1-3, 12-18, 27-33, 42-48, 57-63, 72-78, 87-90, 92, and 95-98 (renumbered as 1-47)*.

3. ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
   a) ☐ All b) ☐ Some* c) ☐ None of the:
      1. ☐ Certified copies of the priority documents have been received.
      2. ☐ Certified copies of the priority documents have been received in Application No. _____ .
      3. ☐ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

   * Certified copies not received: _____.

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.
THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.

4. ☐ A SUBSTITUTE OATH OR DECLARATION must be submitted. Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL PATENT APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient.

5. ☒ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
   (a) ☐ including changes required by the Notice of Draftsperson's Patent Drawing Review (PTO-948) attached
      1) ☐ hereto or 2) ☐ to Paper No./Mail Date _____.
   (b) ☒ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date _____.

Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).

6. ☐ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Attachment(s)

1. ☐ Notice of References Cited (PTO-892)
2. ☐ Notice of Draftperson's Patent Drawing Review (PTO-948)
3. ☒ Information Disclosure Statements (PTO/SB/08), Paper No./Mail Date 1/16/2007
4. ☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material
5. ☐ Notice of Informal Patent Application
6. ☒ Interview Summary (PTO-413), Paper No./Mail Date *20070411* .
7. ☒ Examiner's Amendment/Comment
8. ☐ Examiner's Statement of Reasons for Allowance
9. ☒ Other *See Continuation Sheet*.

MENGAL T. AN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100

Continuation of Attachment(s) 9. Other: Applicant's emailed amendment attatched to this Examiner's Amendment

EXAMINER'S AMENDMENT

1. An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

Authorization for this examiner's amendment was given in a telephone interview with Mr. Kevin Zilka (reg. 41,429) on 4/11/2007.

The application has been amended as follows:

This examiner's amendment is created from the emailed amendment attached hereto, received by the examiner from the applicant. See the attachment.

Mr. Zilka agreed to submit the formal drawings in compliance with 37 CFR 1.121(d) as requested by the examiner during the telephone interview.

2. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Insun Kang whose telephone number is 571-272-3724. The examiner can normally be reached on M-R 6:30-5 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, MENG AI AN can be reached on 571-272-3756. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

IK
AU 2193

MENG-AL T. AN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100

Requested

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:	)
	)
Nedbal et al.	) Art Unit: 2193
	)
Application No. 10/091,415	) Examiner: Kang, Insun
	)
Filed: 03/07/2002	) Date: 04/11/2007
	)
For: PROTOCOL FOR CONTROLLING AN EXECUTION PROCESS ON A DESTINATION COMPUTER FROM A SOURCE COMPUTER	)

CERTIFICATE OF FACSIMILE

I hereby certify that this correspondence is being facsimile transmitted to the Commissioner for Patents, Alexandria, VA 22313-1450 at facsimile number: (571) 273-8300 on the above date.

Signed: _____
Erica L. Farlow

AMENDMENT E

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Examiner:

In response to the Office Action mailed 09/22/2006, please enter the following amendments believed to place the claims in condition for allowance.

Requested

IN THE SPECIFICATION:

Please amend the paragraphs spanning lines 10-25 on page 2, as follows:

It is known to provide various data transmission protocols that allow remote procedure calls and responses. One example of such a known protocol is SOAP (Simple Object Access Protocol) that is based on XML and defines a protocol for the exchange of information in the form of XML. This includes the provision of remote procedure calls and responses via the SOAP RPC protocol ~~(for example see http://www.w3.org/TR/soap12-part1)~~. A significant disadvantage of SOAP is that it is closely related to http. This in turn leads to the requirement for an http server equipped with a SOAP capability on each computer involved in a SOAP remote procedure call. This is a significant cost, complexity, and performance overhead.

Another known protocol for initiating remote procedure calls is that XML-RPC protocol ~~(for example see http://wwww.xmlrpc.com)~~. A disadvantage of this protocol is that the software architecture at the destination computer combines the process that receives the transmitted data together with the execution processes that are to be triggered in an inflexible manner that is not readily extensible and may not be easily user customised.

Requested

IN THE CLAIMS

Amended claims follow:

1. (Currently Amended) A computer program product embodied on a tangible computer [[readable]]storage medium executed by a computer for triggering an operation at a destination computer using data transferred between a source computer and said destination computer, said computer program product comprising:

receiving code to receive at said destination computer operation specifying XML data sent by said source computer;

parsing code to parse said operation specifying XML data to identify one or more complex data types within said operation specifying XML data;

matching code to match each complex data type with an associated execution process available to said destination computer;

triggering code to trigger processing by [[the or]]each execution process associated with a complex data type within said operation specifying XML data; and

validating code to validate said operation specifying XML data received at said destination computer against schema data, where said schema data is sent to said destination computer from said source computer at the same time as said operation specifying XML data;

wherein said operation performed includes configuring said destination computer to execute a computer program;

wherein said execution process maps configuration data specified within said operation specifying XML data to a configuration data store of said destination computer;

wherein said configuration data store is one of:

a Windows Registry entry;

an INI file;

a DAPI store; and

a database entry;

wherein an identifier of said execution process within said complex data type includes at least one of:

Requested

data specifying a computer file to trigger said execution process;

data specifying a communication channel to trigger said execution process; and

data specifying an operating system command to trigger said execution process;

wherein said operation includes returning result data from said destination computer to said source computer in dependence upon said operation performed by said execution process;

wherein said result data includes data specifying existing configuration data of said destination computer;

wherein said execution process maps existing configuration data of said destination computer stored within said configuration data store of said destination computer to said result data to be returned to said source computer;

wherein said operation specifying XML data is parsed after validating said operation specifying XML data to extract at least one identifier for mapping said at least one identifier to an available execution process;

wherein said operation specifying XML data includes parameter data used by said execution process in said operation.

2. (Previously Presented) A computer program product as claimed in claim 1, wherein parameter data used by said execution process is represented by data within said complex data type of said execution process.

3. (Original) A computer program product as claimed in claim 1, wherein said operation performed includes making a call to an API available to said destination computer.

4. (Cancelled)

5. (Cancelled)

6. (Cancelled)

7. (Cancelled)

8. (Cancelled)

9. (Cancelled)

10. (Cancelled)

11. (Cancelled)

12. (Previously Presented) A computer program product as claimed in claim 1, wherein said result data is passed from said destination computer to said source computer as XML data.

13. (Original) A computer program product as claimed in claim 1, wherein said operation includes returning result data from said destination computer to said source computer in dependence upon whether or not said execution process is available to said destination computer.

14. (Original) A computer program product as claimed in claim 1, wherein an operation that may be performed by said destination computer includes installing a new execution process.

15. (Original) A computer program product as claimed in claim 1, wherein said operation specifying data is validated by said destination computer by comparing with a template defining valid data.

16. (Currently Amended) A computer program product embodied on a tangible computer [[readable]]storage medium executed by a computer for triggering an operation

Requested

at a destination computer using data transferred between a source computer and said destination computer, said computer program product comprising:

data forming code to form at said source computer operation specifying XML data containing one or more complex data types; and

transmitting code to transmit from said source computer to said destination computer said operation specifying XML data;

wherein [[the or]]each complex data type within said operation specifying XML data corresponds to an execution process available to said destination computer to be triggered to operate;

wherein said operation performed includes configuring said destination computer to execute a computer program;

wherein said execution process maps configuration data specified within said operation specifying XML data to a configuration data store of said destination computer;

wherein said configuration data store is one of:

a Windows Registry entry;

an INI file;

a DAPI store; and

a database entry;

wherein an identifier of said execution process within said complex data type includes at least one of:

data specifying a computer file to trigger said execution process;

data specifying a communication channel to trigger said execution process; and

data specifying an operating system command to trigger said execution process;

wherein said operation includes returning result data from said destination computer to said source computer in dependence upon said operation performed by said execution process;

wherein said result data includes data specifying existing configuration data of said destination computer;

Requested

wherein said execution process maps existing configuration data of said destination computer stored within said configuration data store of said destination computer to said result data to be returned to said source computer;

wherein said operation specifying XML data is parsed after validating said operation specifying XML data to extract at least one identifier for mapping said at least one identifier to an available execution process;

wherein said operation specifying XML data includes parameter data used by said execution process in said operation;

wherein schema data is transmitted from said source computer to said destination computer at the same time as said operation specifying XML data.

17. (Previously Presented) A computer program product as claimed in claim 16, wherein parameter data used by said execution process is represented by data within said complex data type of said execution process.

18. (Original) A computer program product as claimed in claim 16, wherein said operation performed includes making a call to an API available to said destination computer.

19. (Cancelled)

20. (Cancelled)

21. (Cancelled)

22. (Cancelled)

23. (Cancelled)

24. (Cancelled)

requested

25. (Cancelled)

26. (Cancelled)

27. (Previously Presented) A computer program product as claimed in claim 16, wherein said result data is passed from said destination computer to said source computer as XML data.

28. (Original) A computer program product as claimed in claim 16, wherein said operation includes returning result data from said destination computer to said source computer in dependence upon whether or not said execution process is available to said destination computer.

29. (Currently Amended) A computer program product as claimed in claim 16, wherein an operation that may be performed by said destination computer includes installing a new execution process.

30. (Original) A computer program product as claimed in claim 16, wherein said operation specifying data is validated by said destination computer by comparing with a template defining valid data.

31. (Currently Amended) A method of triggering an operation at a destination computer using data transferred between a source computer and said destination computer, said method comprising the steps of:

receiving at said destination computer operation specifying XML data sent by said source computer;

parsing said operation specifying XML data to identify one or more complex data types within said operation specifying XML data;

matching [[the or]]each complex data type with an associated execution process available to said destination computer;

triggering processing by [[the or]]each execution process associated with a complex data type within said operation specifying XML data; and

validating said operation specifying XML data received at said destination computer against schema data, where said schema data is sent to said destination computer from said source computer at the same time as said operation specifying XML data;

wherein said operation performed includes configuring said destination computer to execute a computer program;

wherein said execution process maps configuration data specified within said operation specifying XML data to a configuration data store of said destination computer;

wherein said configuration data store is one of:

a Windows Registry entry;

an INI file;

a DAPI store; and

a database entry;

wherein an identifier of said execution process within said complex data type includes at least one of:

data specifying a computer file to trigger said execution process;

data specifying a communication channel to trigger said execution process; and

data specifying an operating system command to trigger said execution process;

wherein said operation includes returning result data from said destination computer to said source computer in dependence upon said operation performed by said execution process;

wherein said result data includes data specifying existing configuration data of said destination computer;

wherein said execution process maps existing configuration data of said destination computer stored within said configuration data store of said destination computer to said result data to be returned to said source computer;

Requested

wherein said operation specifying XML data is parsed after validating said operation specifying XML data to extract at least one identifier for mapping said at least one identifier to an available execution process;

wherein said operation specifying XML data includes parameter data used by said execution process in said operation.

32. (Previously Presented) A method as claimed in claim 31, wherein parameter data used by said execution process is represented by data within said complex data type of said execution process.

33. (Original) A method as claimed in claim 31 wherein said operation performed includes making a call to an API available to said destination computer.

34. (Cancelled)

35. (Cancelled)

36. (Cancelled)

37. (Cancelled)

38. (Cancelled)

39. (Cancelled)

40. (Cancelled)

41. (Cancelled)

42. (Previously Presented) A method as claimed in claim 31, wherein said result data is passed from said destination computer to said source computer as XML data.

Requested

43. (Previously Presented) A method as claimed in claim 31, wherein said operation includes returning result data from said destination computer to said source computer in dependence upon whether or not said execution process is available to said destination computer.

44. (Previously Presented) A method as claimed in claim 31, wherein an operation that may be performed by said destination computer includes installing a new execution process.

45. (Previously Presented) A method as claimed in claim 31, wherein said operation specifying data is validated by said destination computer by comparing with a template defining valid data.

46. (Currently Amended) A method of triggering an operation at a destination computer using data
transferred between a source computer and said destination computer, said method comprising the steps of:

forming at said source computer operation specifying XML data containing one or more complex data types; and

transmitting from said source computer to said destination computer said operation specifying XML data;

wherein [[the or]]each complex data type within said operation specifying XML data corresponds to an execution process available to said destination computer to be triggered to operate;

wherein said operation performed includes configuring said destination computer to execute a computer program;

wherein said execution process maps configuration data specified within said operation specifying XML data to a configuration data store of said destination computer;

wherein said configuration data store is one of:

a Windows Registry entry;

an INI file;

a DAPI store; and

a database entry;

wherein an identifier of said execution process within said complex data type includes at least one of:

data specifying a computer file to trigger said execution process;

data specifying a communication channel to trigger said execution process; and

data specifying an operating system command to trigger said execution process;

wherein said operation includes returning result data from said destination computer to said source computer in dependence upon said operation performed by said execution process;

wherein said result data includes data specifying existing configuration data of said destination computer;

wherein said execution process maps existing configuration data of said destination computer stored within said configuration data store of said destination computer to said result data to be returned to said source computer;

wherein said operation specifying XML data is parsed after validating said operation specifying XML data to extract at least one identifier for mapping said at least one identifier to an available execution process;

wherein said operation specifying XML data includes parameter data used by said execution process in said operation;

wherein schema data is transmitted from said source computer to said destination computer at the same time as said operation specifying XML data.

47. (Previously Presented) A method as claimed in claim 46, wherein parameter data used by said execution process is represented by data within said complex data type of said execution process.

Requested

48. (Original) A method as claimed in claim 46, wherein said operation performed includes making a call to an API available to said destination computer.

49. (Cancelled)

50. (Cancelled)

51. (Cancelled)

52. (Cancelled)

53. (Cancelled)

54. (Cancelled)

55. (Cancelled)

56. (Cancelled)

57. (Previously Presented) A method as claimed in claim 46, wherein said result data is passed from said destination computer to said source computer as XML data.

58. (Original) A method as claimed in claim 46, wherein said operation includes returning result data from said destination computer to said source computer in dependence upon whether or not said execution process is available to said destination computer.

59. (Original) A method as claimed in claim 46, wherein an operation that may be performed by said destination computer includes installing a new execution process.

Requested

60. (Original) A method as claimed in claim 46, wherein said operation specifying data is validated by said destination computer by comparing with a template defining valid data.

61. (Currently Amended) Apparatus for triggering an operation at a destination computer using data transferred between a source computer and said destination computer, said apparatus
comprising:

receiving logic to receive at said destination computer operation specifying XML data sent by said source computer;

parsing logic to parse said operation specifying XML data to identify one or more complex data types within said operation specifying XML data;

matching logic to match [[the or]]each complex data type with an associated execution process available to said destination computer;

triggering logic to trigger processing by [[the or]]each execution process associated with a complex data type within said operation specifying XML data; and

validating logic to validat[[ing]]e said operation specifying XML data received at said destination computer against schema data, where said schema data is sent to said destination computer from said source computer at the same time as said operation specifying XML data;

wherein said operation performed includes configuring said destination computer to execute a computer program;

wherein said execution process maps configuration data specified within said operation specifying XML data to a configuration data store of said destination computer;

wherein said configuration data store is one of:

a Windows Registry entry;

an INI file;

a DAPI store; and

a database entry;

wherein an identifier of said execution process within said complex data type includes at least one of:

Requested

data specifying a computer file to trigger said execution process;

data specifying a communication channel to trigger said execution process; and

data specifying an operating system command to trigger said execution process;

wherein said operation includes returning result data from said destination computer to said source computer in dependence upon said operation performed by said execution process;

wherein said result data includes data specifying existing configuration data of said destination computer;

wherein said execution process maps existing configuration data of said destination computer stored within said configuration data store of said destination computer to said result data to be returned to said source computer;

wherein said operation specifying XML data is parsed after validating said operation specifying XML data to extract at least one identifier for mapping said at least one identifier to an available execution process;

wherein said operation specifying XML data includes parameter data used by said execution process in said operation.

62. (Previously Presented) Apparatus as claimed in claim 61, wherein parameter data used by said execution process is represented by data within said complex data type of said execution process.

63. (Original) Apparatus as claimed in claim 61, wherein said operation performed includes making a call to an API available to said destination computer.

64. (Cancelled)

65. (Cancelled)

66. (Cancelled)

Requested

67. (Cancelled)

68. (Cancelled)

69. (Cancelled)

70. (Cancelled)

71. (Cancelled)

72. (Previously Presented) Apparatus as claimed in claim 61, wherein said result data is passed from said destination computer to said source computer as XML data.

73. (Original) Apparatus as claimed in claim 61, wherein said operation includes returning result data from said destination computer to said source computer in dependence upon whether or not said execution process is available to said destination computer.

74. (Original) Apparatus as claimed in claim 61, wherein an operation that may be performed by said destination computer includes installing a new execution process.

75. (Original) Apparatus as claimed in claim 61, wherein said operation specifying data is validated by said destination computer by comparing with a template defining valid data.

76. (Currently Amended) Apparatus for triggering an operation at a destination computer using data transferred between a source computer and said destination computer, said apparatus comprising:

data forming logic to form at said source computer operation specifying XML data containing one or more complex data types;

Requested

transmitting logic to transmit from said source computer to said destination computer said operation specifying XML data;

wherein [[the or]]each complex data type within said operation specifying XML data corresponds to an execution process available to said destination computer to be triggered to operate;

wherein said operation performed includes configuring said destination computer to execute a computer program;

wherein said execution process maps configuration data specified within said operation specifying XML data to a configuration data store of said destination computer;

wherein said configuration data store is one of:

a Windows Registry entry;

an INI file;

a DAPI store; and

a database entry;

wherein an identifier of said execution process within said complex data type includes at least one of:

data specifying a computer file to trigger said execution process;

data specifying a communication channel to trigger said execution process; and

data specifying an operating system command to trigger said execution process;

wherein said operation includes returning result data from said destination computer to said source computer in dependence upon said operation performed by said execution process;

wherein said result data includes data specifying existing configuration data of said destination computer;

wherein said execution process maps existing configuration data of said destination computer stored within said configuration data store of said destination computer to said result data to be returned to said source computer;

Requested

wherein said operation specifying XML data is parsed after validating said operation specifying XML data to extract at least one identifier for mapping said at least one identifier to an available execution process;

wherein said operation specifying XML data includes parameter data used by said execution process in said operation;

wherein schema data is transmitted from said source computer to said destination computer at the same time as said operation specifying XML data.

77. (Previously Presented) Apparatus as claimed in claim 76, wherein parameter data used by said execution process is represented by data within said complex data type of said execution process.

78. (Original) Apparatus as claimed in claim 76, wherein said operation performed includes making a call to an API available to said destination computer.

79. (Cancelled)

80. (Cancelled)

81. (Cancelled)

82. (Cancelled)

83. (Cancelled)

84. (Cancelled)

85. (Cancelled)

86. (Cancelled)

87. (Previously Presented) Apparatus as claimed in claim 76, wherein said result data is passed from said destination computer to said source computer as XML data.

88. (Original) Apparatus as claimed in claim 76, wherein said operation includes returning result data from said destination computer to said source computer in dependence upon whether or not said execution process is available to said destination computer.

89. (Original) Apparatus as claimed in claim 76, wherein an operation that may be performed by said destination computer includes installing a new execution process.

90. (Original) Apparatus as claimed in claim 76, wherein said operation specifying data is validated by said destination computer by comparing with a template defining valid data.

91. (Cancelled)

92. (Previously Presented) A computer program product as claimed in claim 1, further comprising validating said operation specifying XML data received at said destination computer against said schema data, where said schema data is present in said destination computer when said operation specifying XML data is sent.

93. (Cancelled)

94. (Cancelled)

95. (Previously Presented) A computer program product as claimed in claim 1, wherein said validating of said operation specifying XML data and said schema data transmitted from said source computer to said destination computer at the same time generates a validation result.

96. (Previously Presented) A computer program product as claimed in claim 95, wherein said validation result triggers at least one of a valid configuration response and an invalid configuration response.

97. (Previously Presented) A computer program product as claimed in claim 96, wherein said invalid configuration response generates an error message.

98. (Previously Presented) A computer program product as claimed in claim 96, wherein said valid configuration response starts execution of an associated computer program.

REMARKS

Per the telephonic interview conducted with the Examiner on 04/11/2007, applicant has clarified the claims as suggested by the Examiner. Thus, all of the independent claims are deemed allowable. Moreover, the remaining dependent claims are further deemed allowable, in view of their dependence on such independent claims.

In the event a telephone conversation would expedite the prosecution of this application, the Examiner may reach the undersigned at (408) 505-5100. The Commissioner is authorized to charge any additional fees or credit any overpayment to Deposit Account No. 50-1351 (Order No. NAI1P480/01.298.01).

Respectfully submitted,
Zilka-Kotab, PC.

/Kevin Zilka/

Kevin J. Zilka
Registration No. 41,429

P.O. Box 721120
San Jose, CA 95172-1120
408-505-5100